Nama : Dela Fajar Mulia

Kelas : XII RPL 1

No Absen : 07

**Dokumen Dokumentasi Uji LSP**

Skenario : Aplikasi Peminjaman Aset

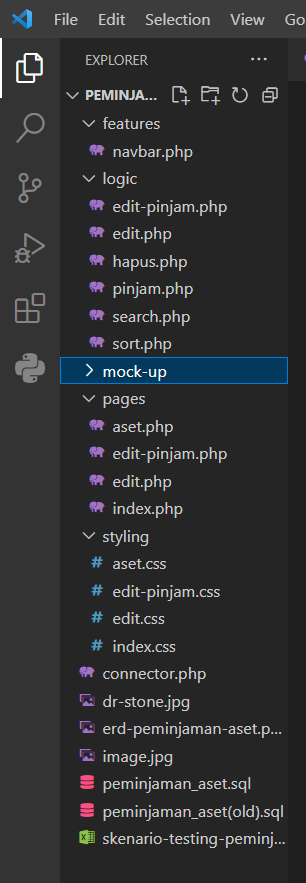
Basis : Website

Bahasa Pemrograman : PHP

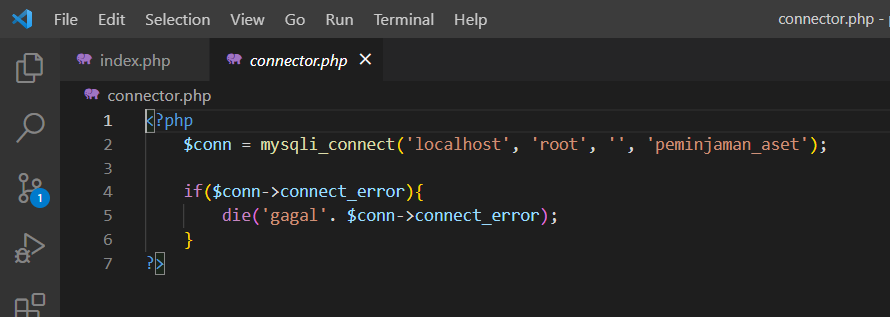
Library : -

Framework : -

**Struktur Folder**

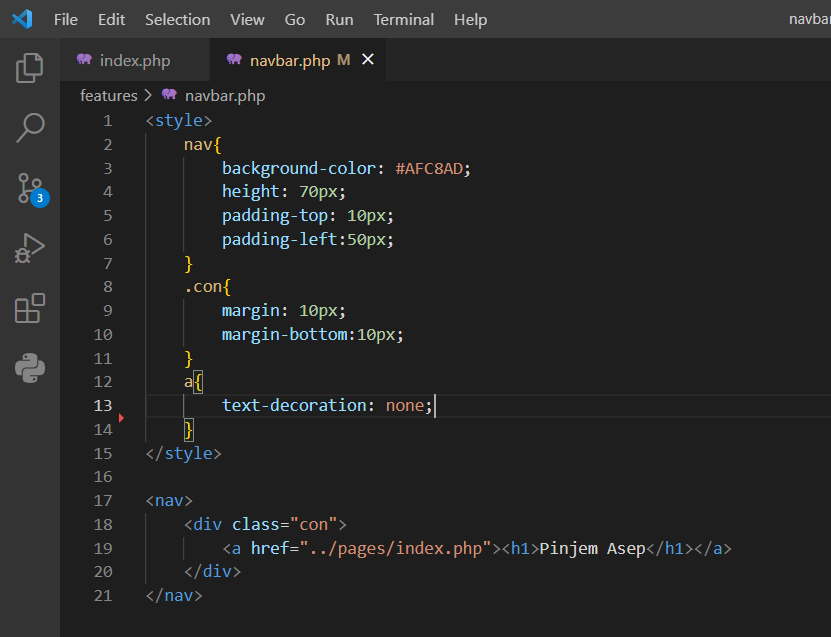
****

**Membuat Koneksi dengan Database**

****

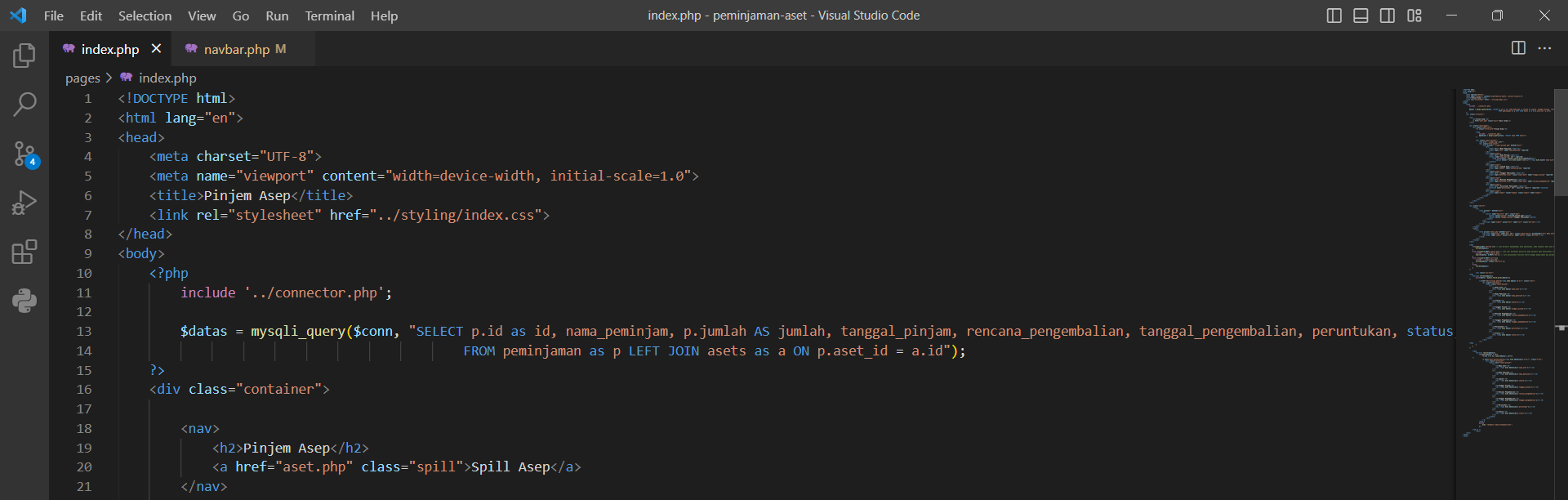
Buat variable bernama conn, yang berisikan query untuk connect dengan database dengan menyertakan hostname, username, password, dan nama database. Apabila koneksi error, maka akan memberikan pesan error.

**Membuat Navbar**

****

Membuat navbar di dalam folder features dengan nama file navbar.php, dengan memberikan tag a untuk menuju halaman index.php yang berada pada folder pages, serta pada halaman ini langsung menggunakan internal CSS.

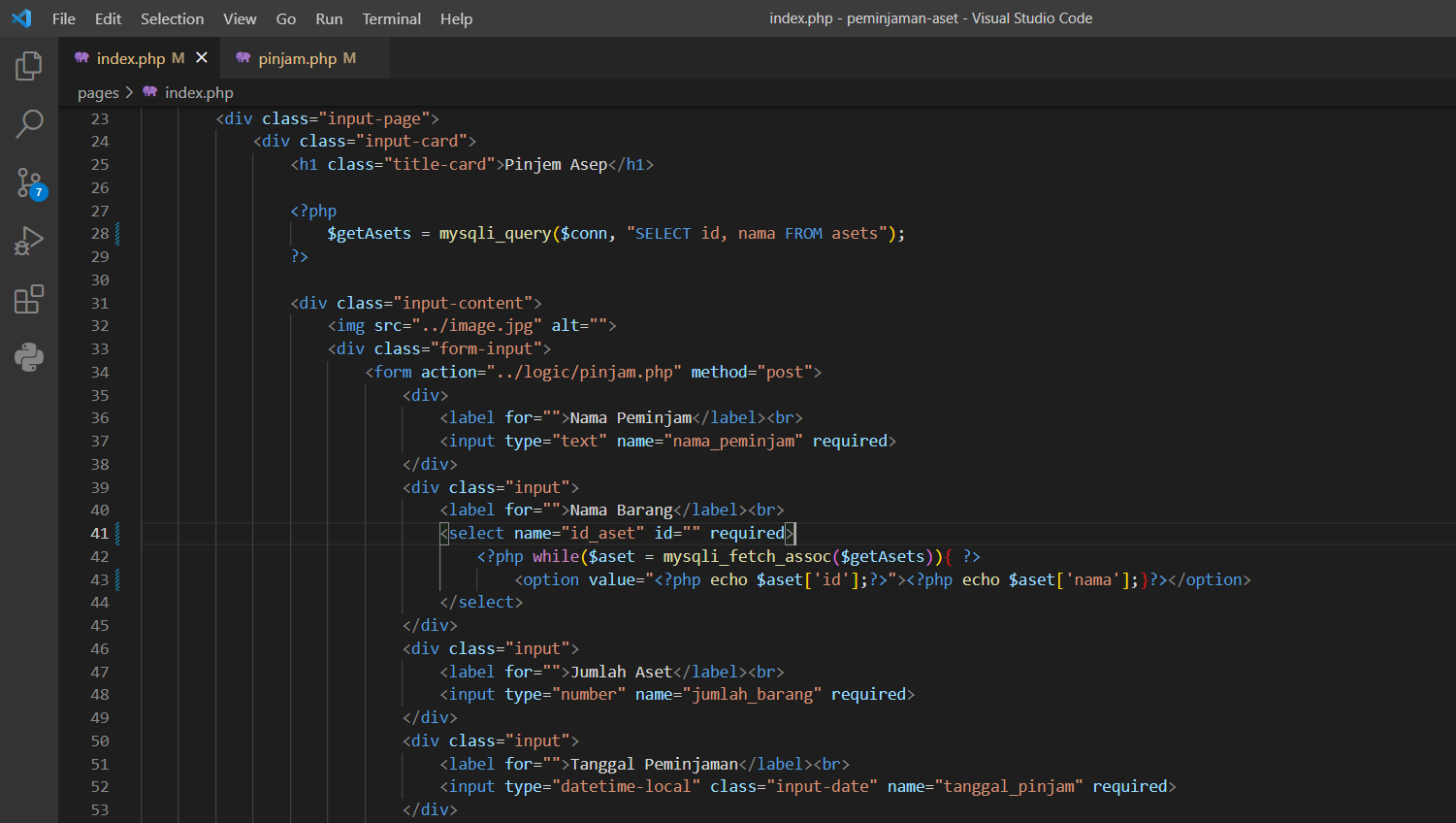
**Mengambil Data untuk Halaman Utama**

****

Pada folder pages terdapat file index.php sebagai halaman utama aplikasi. Dimana saat program pertama kali dijalankan akan meng-include file connector.php sebagai koneksi dengan database yang telah kita buat sebelumnya. Kemudian terdapat variabel dengan nama datas yang berisikan result dari function bawaan PHP yaitu mysqli\_query yang mana perintah tersebut dibuat untuk mengambil beberapa data pada tabel peminjaman yang melakukan left join dengan tabel asets, dimana tabel peminjaman dengan kolom aset\_id itu mereferensikan pada tabel aset kolom id.

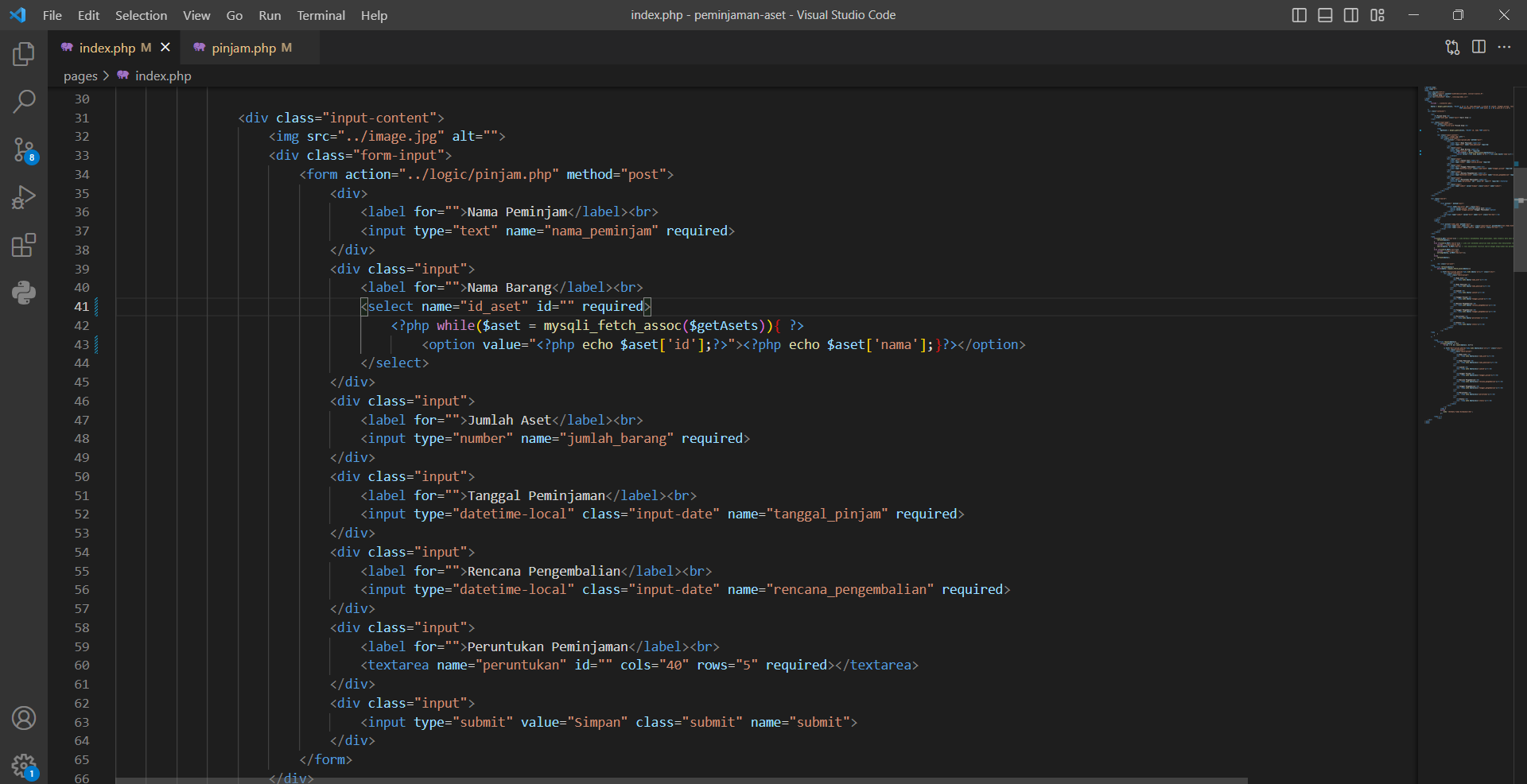
Kemudian terdapat navbar dengan link untuk menuju halaman aset.php.

**Mengambil Nama Aset yang Tersedia**

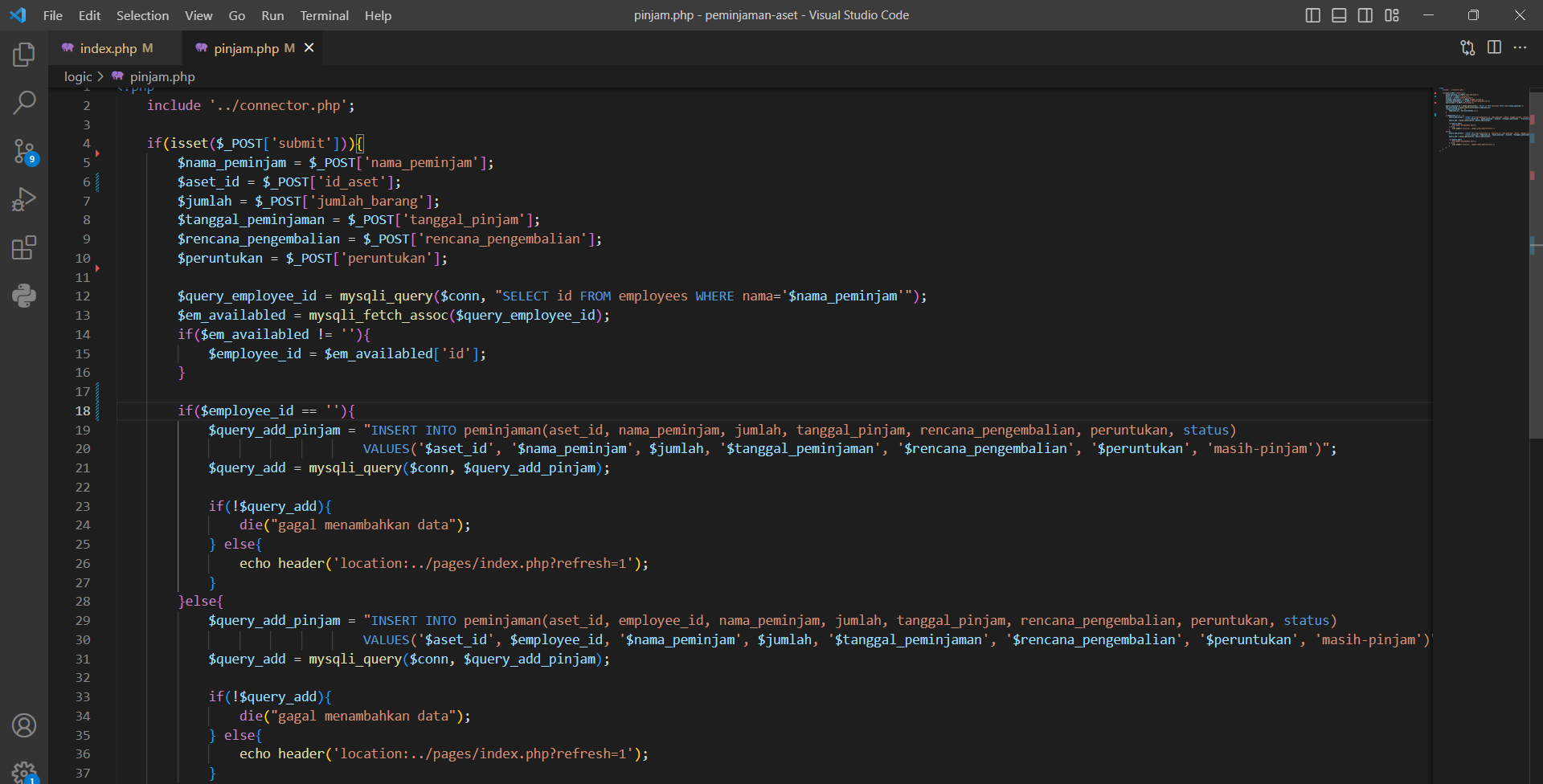
****

Masih di dalam file index.php, terdapat variabel getAsets untuk menyimpan semua id dan nama aset yang tersedia pada tabel aset. Kemudian isi variabel tersebut di looping menggunakan perulangan while, dan masing-masing perulangan disimpan pada variabel aset dengan menggunakan perintah bawaan PHP yaitu mysqli\_fetch\_assoc, dan kemudian ditampilkan pada option dengan value berupa id, dan yang nampak pada user adalah nama aset.

**Melakukan Peminjaman Aset**

****

Pada file index.php terdapat pula form dengan method post dan action yang menuju pada folder logic yang sejajar dengan folder pages, didalam folder logic terdapat file pinjam.php, untuk melakukan peminjaman dengan menginputkan beberapa data yang dibutuhkan seperti nama peminjaman, jumlah aset yang dipinjam, rencana pengembalian, tanggal peminjaman, peruntukan serta nama aset dengan metode yang telah dijelaskan diatas.

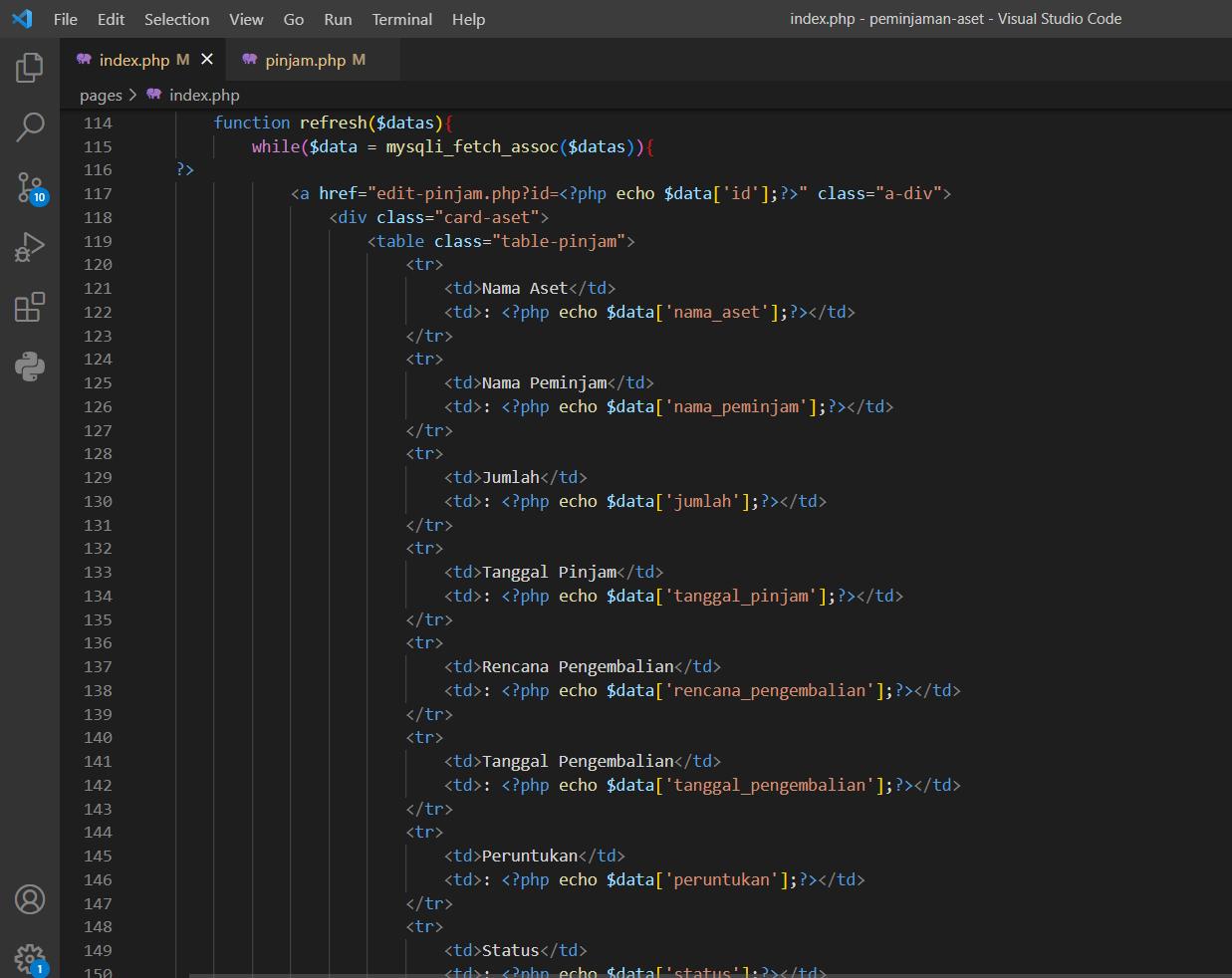


Pada folder logic terdapat file pinjam.php sebagai logika utama dari form peminjaman. Ketika halaman ini ter-reload maka akan langsung meng-include file connector.php yang telah kita buat sebelumnya. Disini akan melakukan pengecekan menggunakan if statement, dengan menggunakan function bawaan PHP isset dicek, apabila user telah mengirimkan pesan ‘submit’ dengan cara mengklik button pinjam pada halaman index.

Kemudian akan mengisikan beberapa variabel yang datanya didapatkan dari apa yang user inputkan dengan cara $namaVar = $\_POST[‘nama-input’];. Dari beberapa variabel yang dibutuhkan diambillah variabel yang menyimpan nama peminjam, nama tersebut akan dicocokkan dengan nama employee yang ada pada tabel employee, jika user yang meminjam adalah employee maka id dari employee tersebut akan disimpan pada variabel employee\_id.

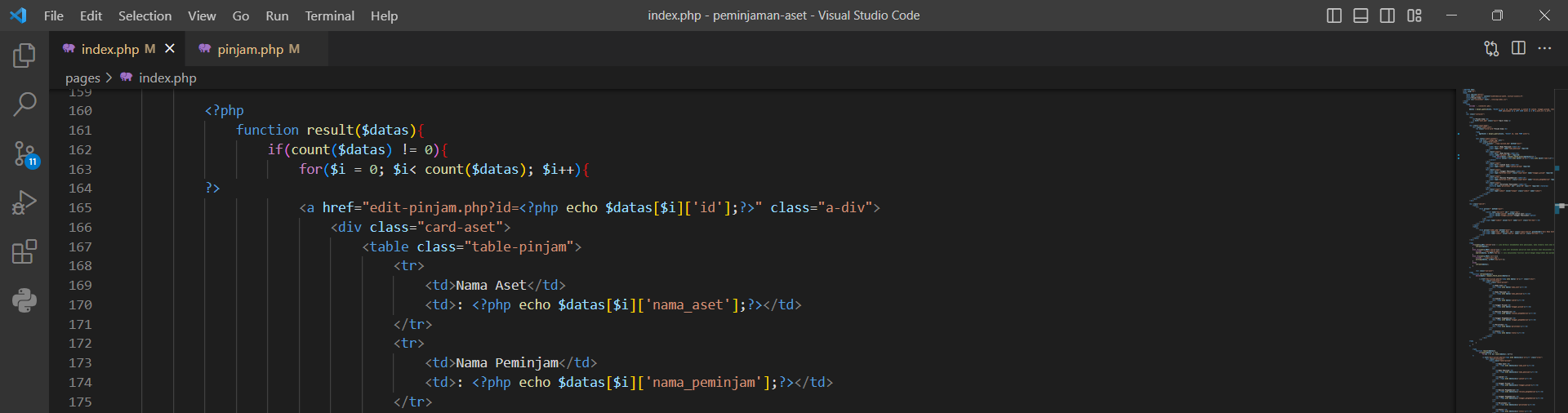
Selanjutnya akan dilakukan pengecekkan, apabila user yang meminjam bukan employee maka data akan disimpan pada tabel peminjaman dengan employee\_id berisikan null, semua kolom akan diisi sesuai dengan yang user inputkan, status peminjaman akan langsung diisi ‘masih-pinjam’, sedangkan kolom pengembalian isinya masih null. Namun apabila user yang meminjam adalah employee maka algoritmanya akan sama dengan sebelumnya, hanya saja kolom employee\_id akan berisikan id dari employee yang pinjam.

**Function Refresh**

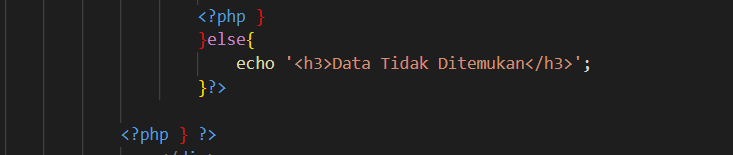
****

Function refresh menerima satu buah parameter dengan nama datas, kemudian dari parameter tersebut akan dilakukan perulangan menggunakan while, setiap kali perulangan data akan disimpan pada variabel bernama data dan akan ditampilkan satu persatu sesuai dengan apa yang dibutuhkan dengan cara <?php echo $data[‘nama\_kolom’];?>

**Function Result**



Function result menerima satu buah parameter dengan nama datas, yang mana nantinya parameter tersebut akan dicek panjangnya, apabila panjangnya tidak sama dengan nol, maka akan dilakukan perulangan sebanyak panjang dari data tersebut, kemudian data tersebut akan ditampilkan dengan cara <?php echo $data[$i][‘nama\_kolom’]; ?>

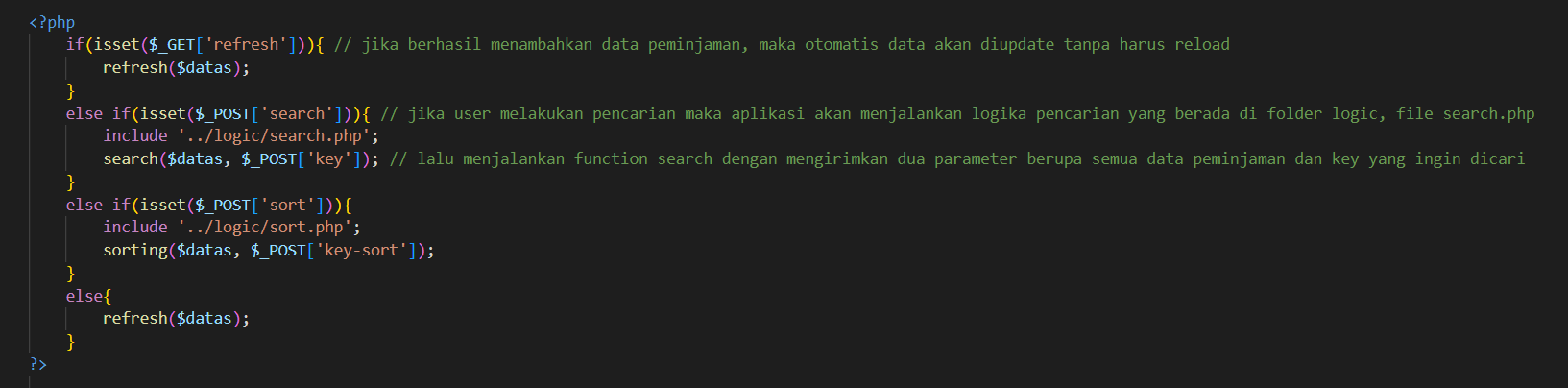


Namun apabila data tidak memenuhi statement pada if, maka akan menampilkan peringatan bahwa data tidak ditemukan.

**Fitur Pengurutan (Sorting)**



Pada halaman index terdapat pula fitur untuk pengurutan data, pengurutan data dapat dilakukan berdasarkan jumlah aset yang dipinjam dan juga tanggal peminjaman.



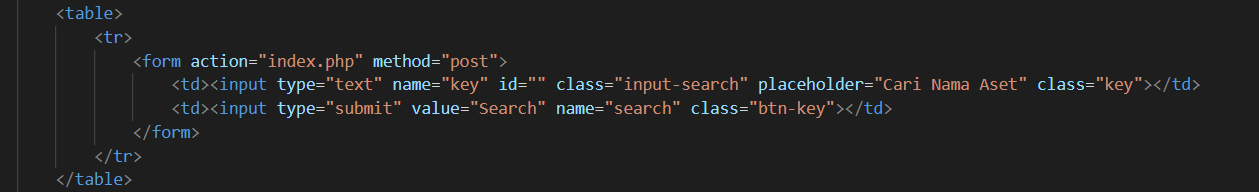
Apabila user mengklik button pengurutan maka program akan menjalankan function sorting yang berada pada folder logic dengan nama file sort.php, dengan mengirimkan dua buah parameter yaitu datas yang telah kita dapatkan sebelumnya, dan juga key yang ingin dicari oleh user.



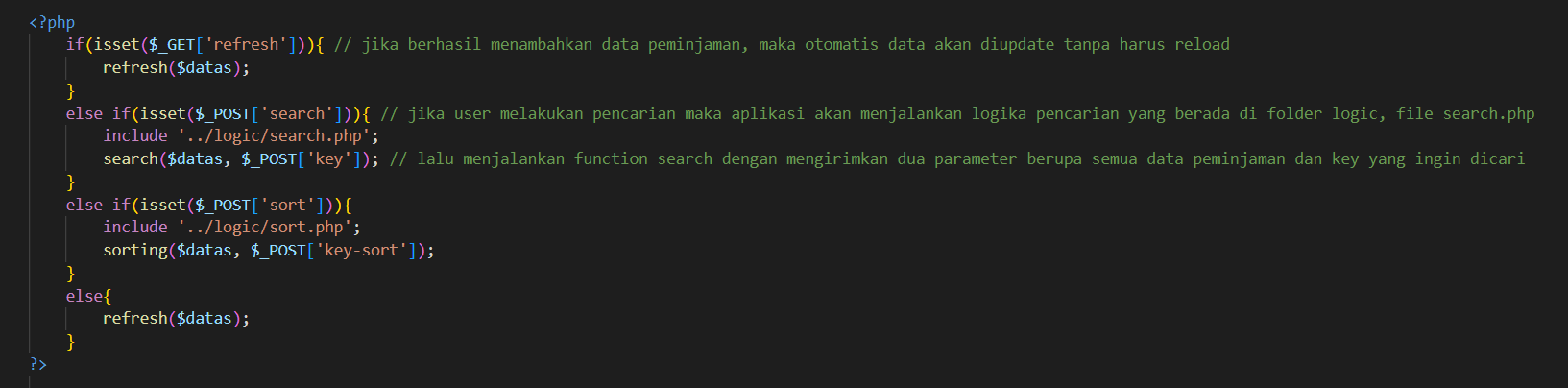
Pada file sort.php program akan mengeksekusi function sorting yang menggunakan algoritma bubble sort dengan mengolah parameter pertama yaitu datas untuk dilakukan fetching kemudian disimpan pada variabel result, kemudian isi dari variabel result dipindahkan kedalam variabel datas dan dihitung panjang dari variabel datas.

Selanjutnya akan dilakukan nested loop untuk melakukan perulangan pada data, apabila data dari index dengan kolom key pada perulangan ke-j lebih besar daripada data dari index dengan kolom key pada perulangan ke-j+1, maka data index ke-j akan disimpan pada variabel temp, kemudian array dengan data index ke-j akan diisi oleh array index ke-j+1, dan isi dari array index ke-j+1 akan diisi oleh array index ke-j, begitu seterusnya hingga semua data sudah terurut. Jika sudah maka function akan meng-include file index.php yang berada pada folder page dan memanggil function result dengan mengirimkan satu parameter berupa array datas. Cara kerja function result telah dijelaskan sebelumnya.

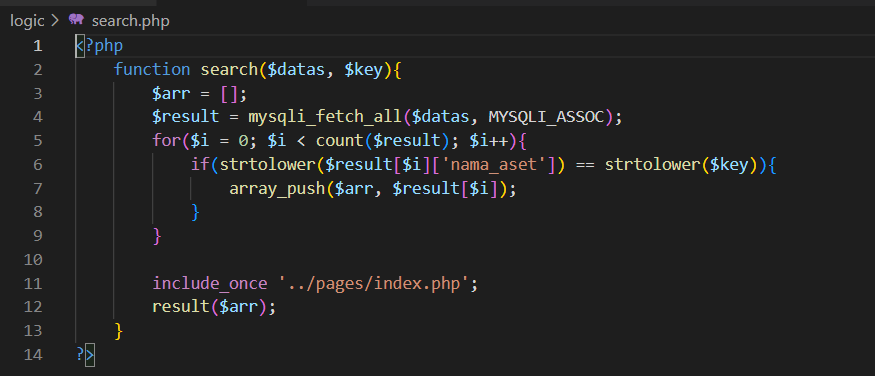
**Fitur Pencarian (Searching)**



Pada halaman index terdapat pula fitur untuk pencarian data, pencarian data dilakukan berdasarkan nama aset yang dipinjam.

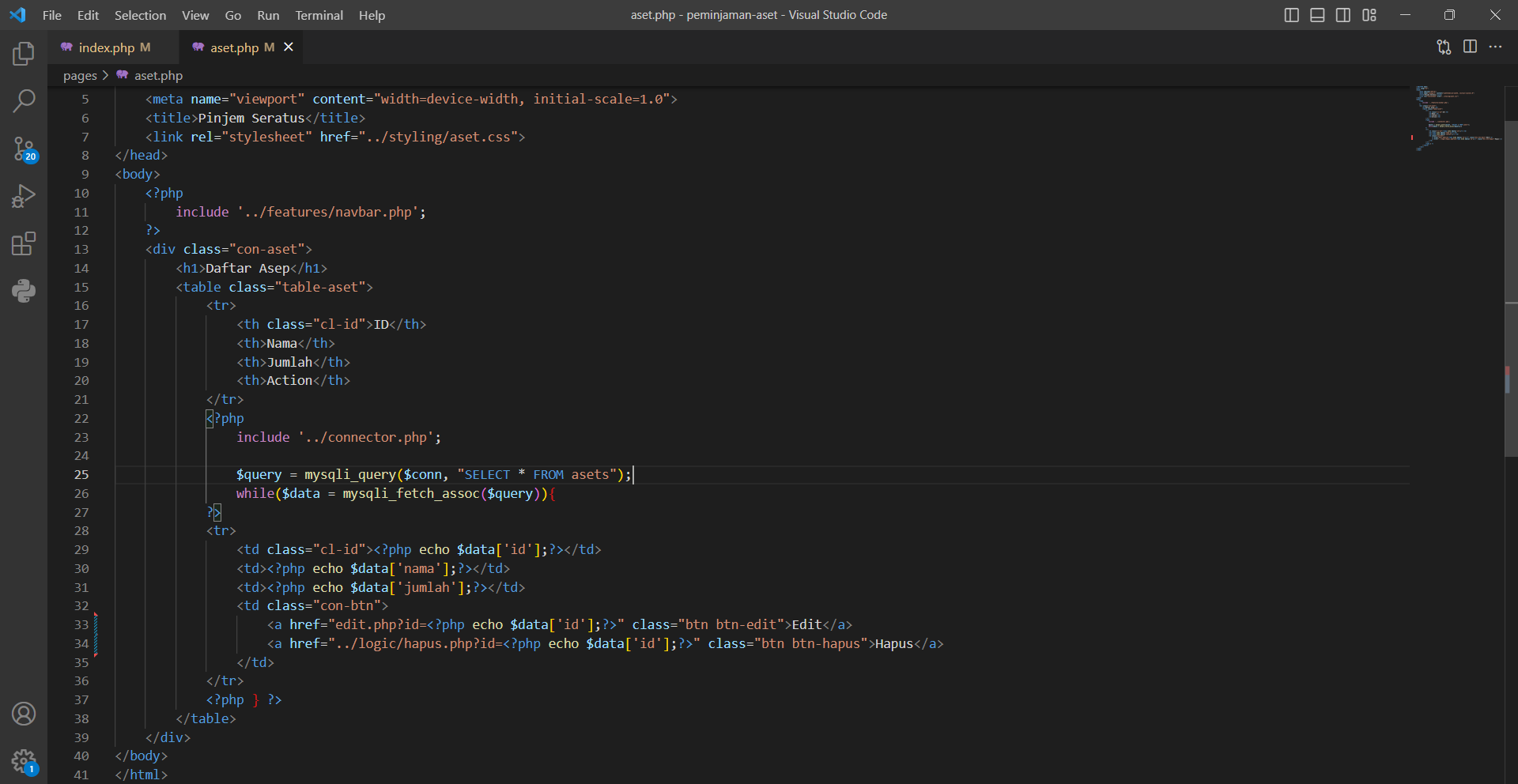


Apabila user mengklik button pencarian maka program akan menjalankan function search yang berada pada folder logic dengan nama file seacrh.php, dengan mengirimkan dua buah parameter yaitu datas yang telah kita dapatkan sebelumnya, dan juga key yang ingin dicari oleh user.



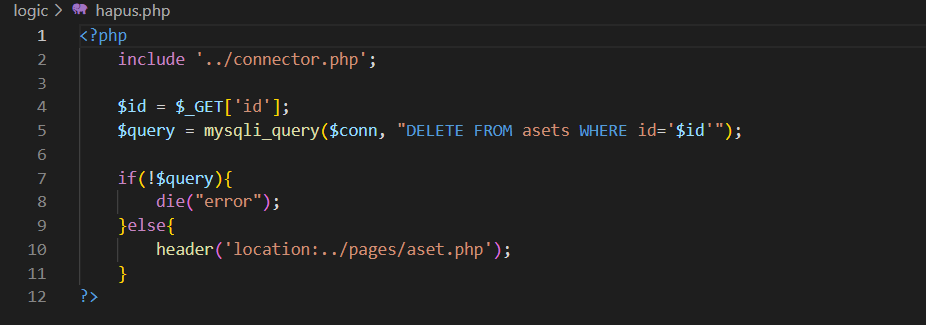
Function search merupakan implementasi dari algoritma linear search, yang mana function ini menerima dua buah parameter. Parameter pertama yaitu datas yang akan dilakukan fetching menggunakan function bawaan dari PHP dan disimpan pada array result, lalu membuat array dengan nama arr dengan isi yang masih kosong. Dari array result kemudian data akan dilakukan perulangan menggunakan for loop, apabila perulangan dari array result dengan index ke i dan kolom nama aset kita bentuk hurufnya menjadi huruf kecil semua menggunakan function bawaan PHP yaitu strtolower sama dengan key pada parameter kedua yang merupakan nama aset yang ingin dicari oleh user dan kita lakukan perubahan hurufnya menjadi huruf kecil semua dengan menggunakan function strtolower, maka kita akan memasukkan data pada result index ke-i kedalam array arr, begitu seterusnya hingga perulangan selesai. Apabila perulangan telah selesai, maka akan meng-include file index.php pada folder pages dan mengeksekusi function result dengan mengirimkan parameter berupa array arr.

**Melihat Daftar Aset**



Halaman aset.php dapat diakses dengan cara mengklik button spill asep yang berada di navbar pada halaman index.php. File ini akan meng-include connector.php, kemudian mengambil semua isi pada kolom tabel asets dan menyimpannya pada variabel query, selanjutnya variabel query akan dilakukan perulangan menggunakan while loop, dan data akan ditampilkan.

**Menghapus Aset**



File ini berada pada folder logic dan akan ter-reload apabila user mengklik salah satu button hapus pada file aset.php. File ini akan meng-include file connector.php, kemudian akan mengambil id yang telah dikirimkan lalu menjalankan perintah untuk menghapus data pada tabel aset sesuai dengan id yang didapatkan, apabila berhasil menghapus data maka akan me-reload halaman aset.php pada folder pages, apabila terjadi error saat akan menghapus data, maka akan menghentikan eksekusi selanjutnya dengan meninggalkan pesan error.

**Mengedit Aset**



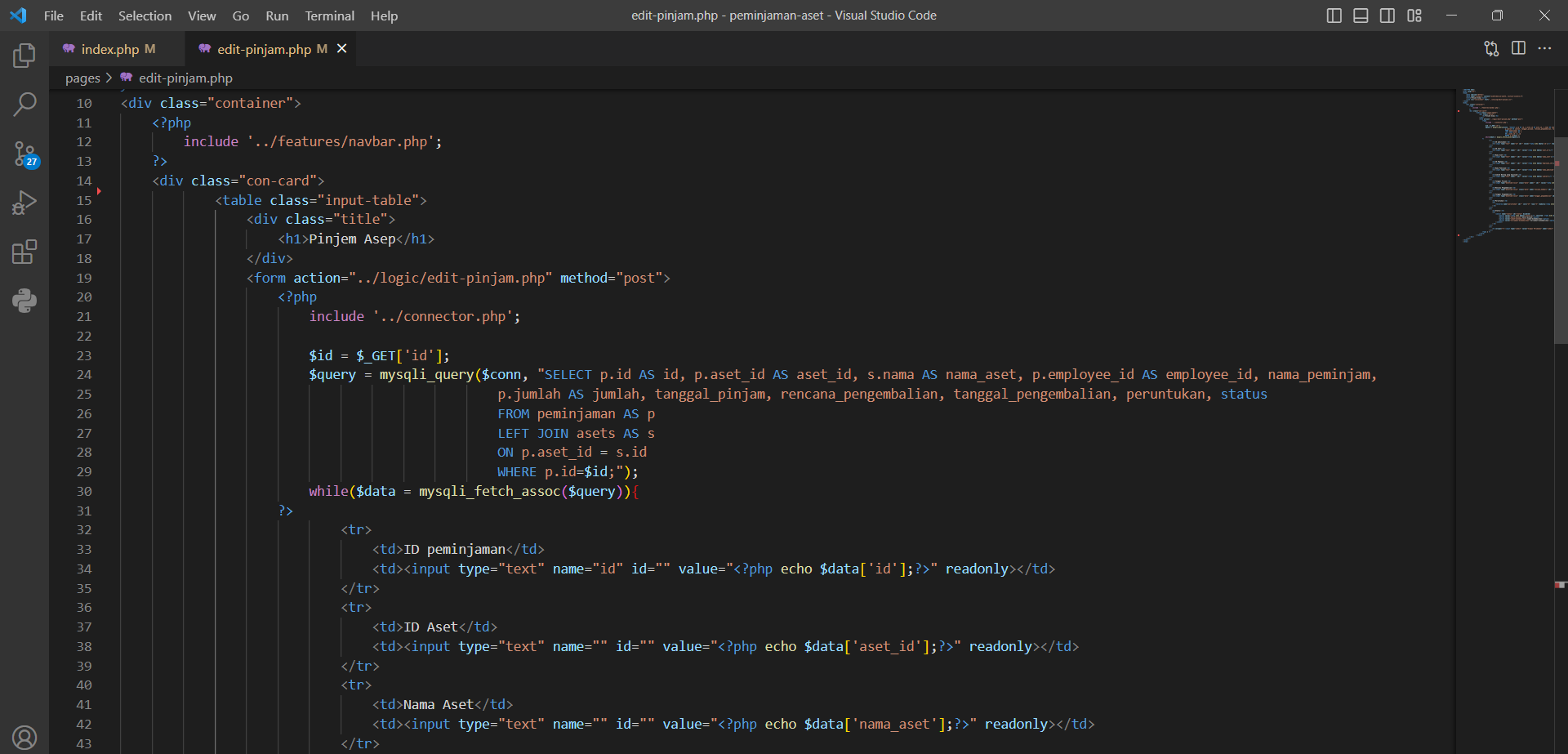
File ini akan ter-reload apabila user mengklik button edit aset pada halaman aset.php. Program akan meng-include file connector.php, lalu mendapatkan id dengan apa yang dikirimkan oleh file aset.php, dan melakukan query untuk mendapatkan semua data aset sesuai dengan id yang telah didapatkan. Kemudian melakukan perulangan menggunakan while loop dan menampilkan semua data sesuai dengan input yang tersedia, memberikan input dengan tipe hidden untuk menyimpan id namun agar tidak terlihat oleh user, dan juga memberikan button untuk mengedit data.



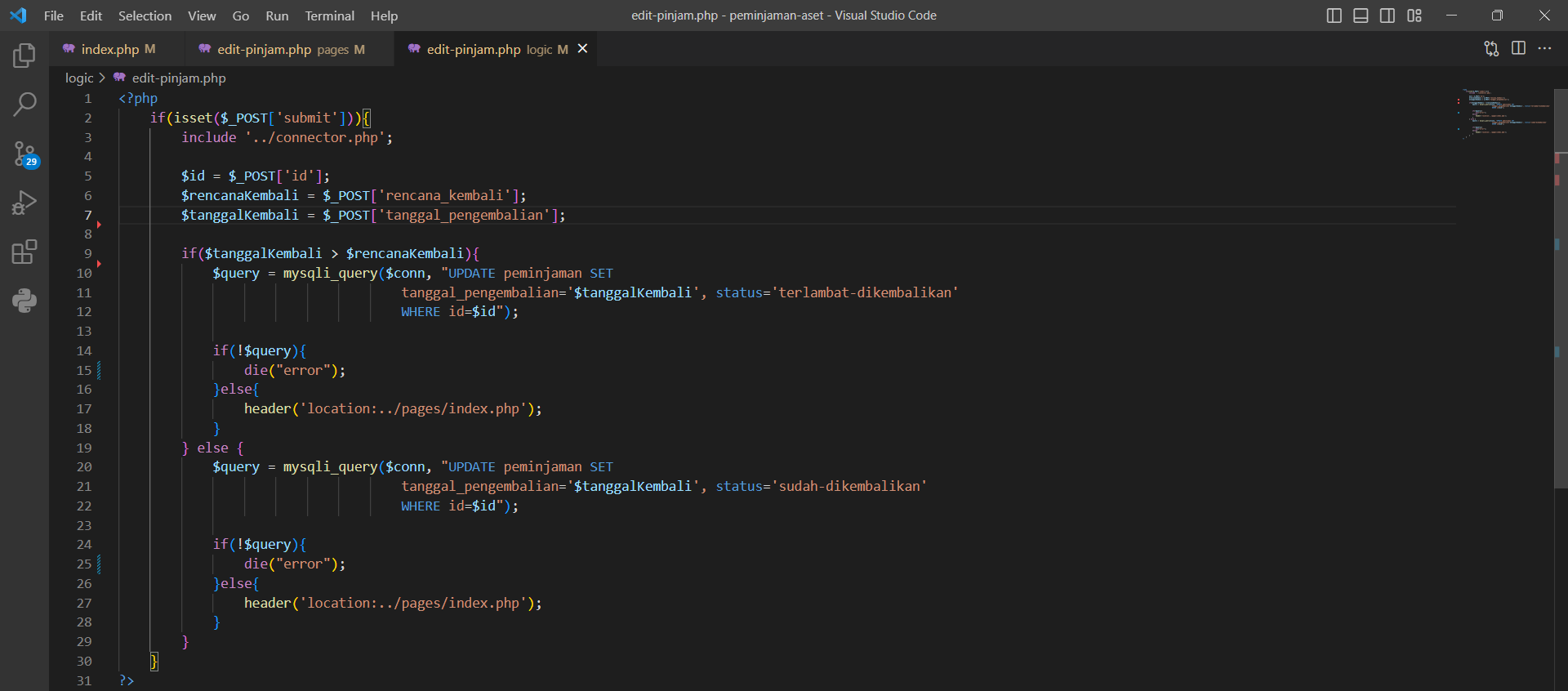
Apabila user mengklik button edit data pada file edit.php maka file edit.php yang berada pada folder logic akan dieksekusi. File ini akan meng-include file connector.php kemudian mengecek apakah button benar sudah diklik, apabila sudah, maka akan menyimpan semua yang telah user inputkan kedalam variabel-variabel yang ada, dan juga variabel oidID untuk mendapatkan id yang disimpan pada input dengan tipe hidden, sedangkan newID digunakan untuk menyimpan apabila terdapat perubahan id.

Lalu akan melakukan query berupa perubahan data pada tabel aset sesuai dengan perubahan yang user lakukan dimana id asetnya sesuai dengan variabel oldID. Apabila query gagal, maka akan menampilkan pesan error. Jika tidak, maka akan menuju halaman aset.php pada folder pages.

**Edit Peminjaman**



Halaman edit-peminjaman.php yang berada pada folder pages akan ter-reload apabila user mengklik salah satu card riwayat peminjaman pada file index.php. Halaman ini akan meng-include connector.php kemudian mendapatkan id sesuai dengan yang telah dikirimkan oleh pada card, dan mengambil beberapa data dari tabel peminjaman, aset dan employee sesuai dengan id peminjaman yang telah didapatkan, dan meloopingnya menggunakan while loop kemudian menampilkan data-datanya sesuai dengan input yang telah disediakan, akan tetapi semuanya hanyalah readonly atau tidak dapat diedit oleh user, kecuali pada input tanggal pengembalian dengan tipe input date.



Apabila user telah melakukan perubahan dan mengklik button simpan perubahan pada file edit peminjaman, maka akan me-reload halaman edit-peminjaman.php pada folder logic. Program akan mengecek apabila user benar sudah mengklik button dengan cara mengecek apakah data dengan name submit sudah tersedia menggunakan isset. Jika sudah, maka program akan meng-include file connector.php dan menyimpan id, rencana pengembalian dan tanggal pengembalian pada masing-masing variabel. Kemudian program akan mengecek menggunakan if statement. Apabila tanggal pengembalian lebih besar daripada tanggal rencana pengembalian, maka program akan mengupdate pada tabel peminjaman dengan status ‘terlambat-dikembalikan’ dimana id peminjaman sesuai dengan id yang telah disimpan pada variabel id, jika query gagal maka akan menampilkan error, namun jika berhasil maka program akan me-reload file index.php pada folder pages. Namun, apabila tanggal pengembalian lebih kecil daripada tanggal rencana pengembalian, maka program akan mengupdate pada tabel peminjaman dengan status ‘sudah-dikembalikan’ dimana id peminjaman sesuai dengan id yang telah disimpan pada variabel id, jika query gagal maka akan menampilkan error, namun jika berhasil maka program akan me-reload file index.php pada folder pages.